A1

Referring to FIG. 1, a block diagram of a method 100 is shown in accordance with a preferred embodiment of the present invention. Fuses in a design may be enumerated and data collected and stored in a file. The data may include schematic path data, verilog simulation path data and/or physical fuse location data. Multiple representations of the design and/or data may be accessible by other tools, such as netlister and/or layout versus schematic (LVS). The data may be used to enumerate the fuses. The data may be implemented in order to test and/or repair the design. References to "verilog" refer to the verilog hardware description language (HDL) as defined by the IEEE 1364-1995 standard.

Please replace the paragraphs starting on page 4, line 7 with the following paragraphs:

A2

FIG. 2 is a block diagram of an example apparatus implementing the present invention; and

FIG. 3 is a flow chart of an operation of the method of FIG. 1.

Please replace the paragraph starting on page 5, line 14 with the following paragraph:

A3

The method 100 may comprise a design data block 102, a netlist block 104, a simulation block 106, a generation block 108, a table block 110, an application block 112, a program statement block 114, a location block 116, a schematic/simulation block 118, a repair program or repair block 120 and a test block 122. The generation block 108 may receive fuse data from the design data block 102. The generation block 108 may write the fuse data into a file and perform error checking on the file.

Please replace the paragraphs starting on page 9, line 10 with the following paragraphs:

A4

Construction of the repair program 120 may rely on part-specific redundancy information and errors found after a first-silicon delay part production. The repair program 120 may receive the errors from the defect block 122. The repair program 120 may be exercised in advance of the first-silicon for specific part failures. The repair program 120 may predict fuse locations that, if programmed, may correct a part experiencing failure. The method 100 may provide an easy and reliable method to perform simulations that emulate a design as if those locations were programmed on the die by the laser.

Referring to FIG. 2, a block diagram of an example apparatus 200 implementing the present invention is shown. The circuit 200 may comprise a design flow block or design flow circuit 202 and a stand-alone block or stand-alone circuit 204. The design flow block or circuit 202 may comprise a fuse network block 206, a fuse LVS block 208 and a design/database circuit 210. The fuse network block 206 and the fuse LVS block 208 may provide data to the stand-alone block or circuit 204. In one example the design/database circuit 210 may be implemented as a design flow and Opus design database.

Please replace the paragraphs starting on page 11, line 1 with the following paragraphs:

The fuse application block 220 may write one or more ASCII report files. A repair memo, also referred to as a repair file, may be manually assembled from the report files. One or more steps may be manually determined and may be incorporated into the repair memo or file. The repair memo or file may be exercised to predict coordinates to program. The coordinates may be mapped to verilog paths. Simulations of the verilog paths may be performed to verify the expected function.

The design flow circuit 202 may further comprise a browse block 240. The browse block 240 may capture, in the schematic, enough knowledge of the fuses to elevate the fuses to a higher level of abstraction. The user may describe, in application terms,

A5

the desired redundancy event. For example, the redundancy event may comprise replacing column C in quadrant Q. The coordinates to program may be automatically determined. However, the simulation may still be required to verify the function (e.g., to verify that the application specific knowledge captured in the schematic describing specific intent of each fuse is correct).